

Note 3A • Recursion on a List

Refer to **Note 0D** to review recursion and **Note 1B** to review entering a list into the Home screen.

When defining a recursive routine, the value for the last answer can come either from a list or it can be a single number, as in **Note 0D**. If you want to generate two patterns at once or keep track of the term numbers of a sequence, using recursion on a list can be useful. When using recursion on a list, you must refer to the number of the term in the list that you want to use. So, $\text{Ans}(2)$ does not mean the last answer times 2, but rather the value of the second term in the previous list.

To use recursion on a list, enter a list of initial values and press $\boxed{\text{ENTER}}$. The list must be enclosed in braces. Then create a formula line that is a list, with each term being a formula using the value of a term(s) in the original list. Finally, press $\boxed{\text{ENTER}}$ repeatedly to generate a sequence of lists.

The screen here shows a sequence of lists, each containing two terms. The pattern of the first term in each list is a sequence that starts with 1 and increases by 1 each time. This sequence gives an index number for the second term in each list. The pattern of the second terms in the lists is a sequence that starts with 4 and increases by 7 each time. So, for example, {4 25} indicates that 25 is the fourth term in the sequence that begins with 4 and increases by 7 with each new term.

```
{0,4}
Ans(2)      {0 4}
              4
```

```
{1,4}
{Ans(1)+1,Ans(2)+7}
              {1 4}
              {2 11}
              {3 18}
              {4 25}
```

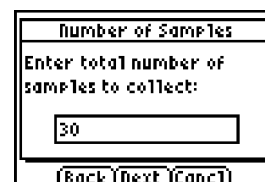
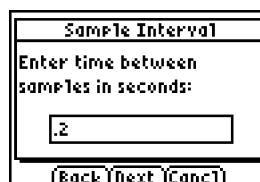
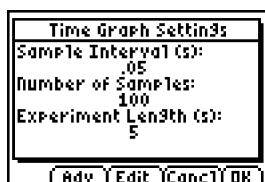
Note 3B/App • Collecting Distance Data Using the EasyData App

You must have a TI-83 Plus or TI-84 Plus to use this Note. If you have a TI-83, see **Note 3B** on pages 32–33. You can also use the programs CBRSET and CBRGET with any calculator. See page vi.

You will need a CBR (Calculator-Based Ranger).

Connect the CBR to the calculator. Press $\boxed{\text{APPS}}$ and select EasyData. The CBR will immediately begin collecting distance data, which is displayed on your calculator screen.

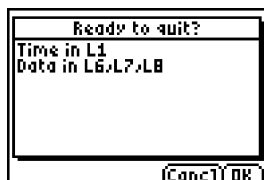
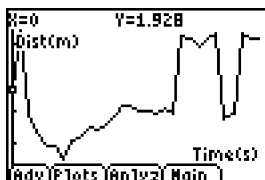
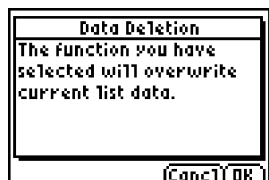
To collect distance data, press Setup ($\boxed{\text{WINDOW}}$) and select 2:Time Graph.... You will be shown the default settings for time interval and number of samples. Press Edit ($\boxed{\text{ZOOM}}$) to edit these settings. Enter 0.2 for the sample interval and press Next. Enter 30 for the number of samples and press Next. Then press OK.



The calculator will resume collecting data. Press Start ($\boxed{\text{ZOOM}}$) to collect the 6 seconds of data you have specified. You will be told that this function will overwrite the current list data. Press OK to continue. The calculator will collect the data and graph it.

(continued)

To end the Application, press Main then press Quit. You will get a message telling you where the data is stored. Time data is in L1, distance data is in L6, velocity data is in L7, and acceleration data is in L8.



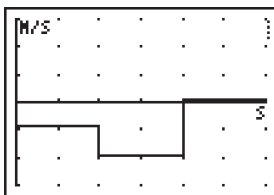
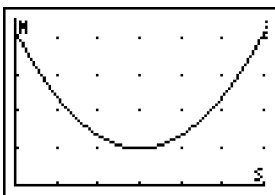
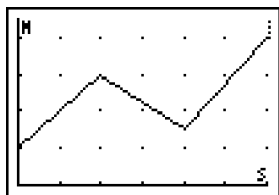
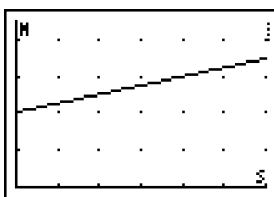
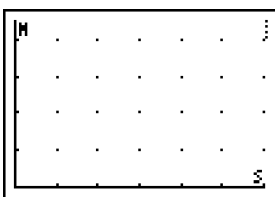
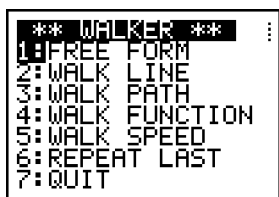
When you press **STAT**, then select 1:Edit, L1 and L6 are shown (you'll need to arrow to the right to see L6.) If you want to see L7 and L8, you'll need to recall those lists. See **Note 1B** for help recalling or moving a list.

Note 3B • WALKER Program

This program can be used with the TI-83, TI-83 Plus, and TI-84 Plus. However, if you have a TI-83 Plus or TI-84 Plus, you should consider using the EasyData App described in **Note 3B/App** on the preceding page.

This program will collect and graph distance-time data or velocity-time data using a motion sensor. Each WALK command allows you to see the graph being constructed as the data are collected. To start, you must select one of the following commands.

- 1:FREE FORM gives you a blank screen for simple data collection.
- 2:WALK LINE draws a random line for you to try to walk.
- 3:WALK PATH draws a random path of three segments for you to match.
- 4:WALK FUNCTION expects you to enter an equation into Y1 *before* you start the program and then to match the path it makes.
- 5:WALK SPEED graphs meters per second and time. You must try to match three horizontal segments.
- 6:REPEAT LAST repeats the previous command on the same graph.



[0, 6, 1, 0, 4.5, 1]

[0, 6, 1, 0, 4.5, 1]

[0, 6, 1, -3, 3, 1]

(continued)

```

PROGRAM:WALKER
Full
ClrHome
Disp "NOW CHECKING THE"
Disp "CALCULATOR-CBR"
Disp "LINK CONNECTION."
Disp "PLEASE WAIT...."
{1,0}→L1
Send(L1)
{0}→L2
Lbl M
{7}→L1
Send(L1)
Get(L2)
If dim(L2)=1 and L2(1)=0
Then
ClrHome
Disp "****LINK ERROR****"
Disp "PUSH IN THE LINK"
Disp "CORD CONNECTORS"
Disp "FIRMLY THEN HIT"
Disp "[ENTER]."
Pause
Goto M
End
Disp ""
Output(6,1," STATUS: O.K.")
Output(8,10,"[ENTER]")
Pause
6→T:4→D:20→R
{47,20,6}→LWDFT
1/R→I
{1,0}→LCLEAR
{1,11,2}→LSONIC
{3,1,-1,0}→LREAD
GridOn:Func
FnOff
Lbl 0:0→F:
PlotsOff 1:FnOff
Menu(" ** WALKER ** ", "FREE
  FORM",7,"WALK LINE",8,"WALK
  PATH",1,"WALK FUNCTION",F,"WALK
  SPEED",2,"REPEAT LAST",3,"QUIT",4)
Lbl F:FnOn 1:99→U
Lbl 7:2→F
Lbl 8:F-1→F
Lbl 1
0→Xmin:T→Xmax
0→Ymin:D+.5→Ymax
1→Xscl:1→Yscl

PlotsOff
randInt(1,20,4)/2→Lq
{0,T/3,2T/3,T}→L3
If F=-1:3L3→L3
Plot2(HyLine,L3,L4,..)
If F=1:PlotsOff
Lbl 5
ClrDraw:DispGraph
Text(0,2,"M")
Text(55,83,"S")
Lbl 9:Pause
Send(LCLEAR)
Send(LSONIC)
Send(LREAD)
int(RT)+1→N
N→dim(L2)
For(J,2,N)
Get(L2(J))
L2(J)→H:If F=2:7(H-L2(J-1))→H
Pt-On(1(J-1),H)
End:Send(LCLEAR)
seq(1J,J,0,N-1)→L1
Pause
Plot1(Scatter,L1,L2,..)
Goto 0
Lbl 2:2→F
GridOff
0→Xmin:-3→Ymin
T→Xmax:3→Ymax
1→Xscl:1→Yscl
PlotsOff
randInt(0,4,6)-1.9→Lq
Lq(1)→Lq(2)
Lq(3)→Lq(4)
Lq(5)→Lq(6)
{0,2,2,4,4,6}→L3
Plot2(HyLine,L3,L4,..)
Lbl 6:2→F
ClrDraw
DispGraph:
Text(0,2,"M/S")
Text(32,83,"S")
Goto 9
Lbl 3
If U=99:Then:FnOn :1→U:End
If Ymin=0:Goto 5
Goto 6
Lbl 4
ClrHome

```

Note 3C • INOUT Program

In this program, you write a linear rule or expression that links a set of input values to their corresponding output values. Before executing the program, choose a level of difficulty that is easy, medium, or hard. We recommend that you choose “EASY” until you get a string of rules correct on the first try. If you make an incorrect guess, the program displays your results and allows you to try again. In the beginning, enter your guess in the form $a + b \cdot L_1$, where a is the starting value and b is the recursive rule. Later you may wish to write your rule in other ways. But always use list L_1 as the variable in your expression. List L_1 is the input list.

```
THIS PROGRAM
WILL DISPLAY TWO
LISTS WITH A
LINEAR RULE.
GUESSES MUST
USE THE VARIABLE
L1 IN THE RULE.
PRESS ENTER.
```

```
INOUT PROGRAM
1:EASY
2:MEDIUM
3:HARD
4:QUIT
```

```
IN OUT
[[0 4 ]
[1 0 ]
[2 -4 ]
[3 -8 ]
[4 -12]
[5 -16]]
GUESS: 4-4L1
```

```
IN OUT YOU ...
[[0 4 4 ]
[1 0 0 ]
[2 -4 -4 ]
[3 -8 -8 ]
[4 -12 -12]
[5 -16 -16]]
RIGHT!
```

```
PROGRAM:INOUT
0→R
ClrHome
Disp "THIS PROGRAM.", "WILL DISPLAY
TWO", "LISTS WITH A ", "LINEAR
RULE. ", "GUESSES MUST", "USE THE
VARIABLE", "L1 IN THE RULE."
Output(8,2,"PRESS ENTER."):Pause
Lbl 0
Menu("INOUT PROGRAM","EASY",1,
"MEDIUM",2, "HARD",3,"QUIT",Q)
Lbl 1
{0,1,2,3,4,5}→L1
randInt(-5,5,2)→L2
Goto 5
Lbl 2
seq(H,H,-3,8)→L1
rand(12)→L2
SortA(L2,L1)
6→dim(L1)
SortA(L1)
randInt(-7,7,2)→L2
Goto 5
Lbl 3
seq(H,H,-9,9,.5)→L1
rand(37)→L2
SortA(L2,L1)
6→dim(L1)
```

```
randInt(-9,9,2)→L2
Lbl 5
ClrHome
L2(1)L1+L2(2)→L2
List►matr(L1,L2,[A])
Disp "IN OUT"
Disp [A]
Repeat sum(L2-L2)=0
If R:Then
Input "START VALUE:",S
Input "CHANGE:",C
S+CL1→L2
Else
Input "GUESS:",L2
End
List►matr(L1,L2,L2,[A])
ClrHome
Disp "IN OUT YOU"
Disp [A]
End
Output(8,3,"RIGHT!")
Pause
Goto 0
Lbl Q:ClrHome
Disp "PRESS ENTER", " TO
REPLAY", " ", " PRESS", " 1 AND
ENTER", " TO QUIT"
```

To enter [A], press **2nd** [MATRIX] and choose [A]; don't use **2nd** [] or **ALPHA** A.

Note 3D • LINES Program

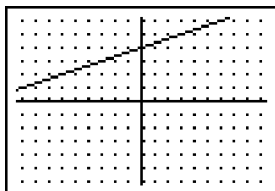
You can use this program to practice writing the equation for a line. Before you begin, clear any equations in the Y= screen, and set the graph style to “line with trail” by moving to the left of Y_1 and pressing **ENTER** four times. When you run the program, the graph of a line will appear. In the “EASY” option, the

(continued)

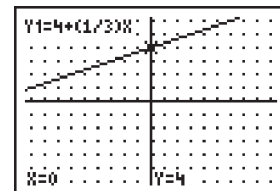
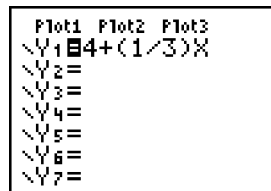
y -intercept will always appear on the screen. In the “HARD” option, the y -intercept may or may not appear on the screen.

Run the program and follow these steps:

- Press $\boxed{1}$ $\boxed{\text{ENTER}}$ to select the “EASY” option, or $\boxed{2}$ $\boxed{\text{ENTER}}$ to select the “HARD” option.
- Press $\boxed{\text{ENTER}}$ to see the graph.
- Study the graph and determine values that will help you write the equation for the line. You can trace on the line to see the coordinates of individual points.
- Press $\boxed{Y=}$ and enter the linear equation that you think matches the line on the graph.
- Press $\boxed{\text{GRAPH}}$ to compare the graph of your line to the program’s line.
- If the two lines don’t match, repeat **steps c, d, and e** until they do. You can enter new equations for other $Y=$ lines or you can clear your old guess before you enter a new one.
- When the graphs match, you will see them as one line. Trace and switch from one line to the other. The equation in the upper-left corner will indicate whether there really are two lines. Pressing $\boxed{\text{ENTER}}$ after you’ve traced a line will only regraph that same line.
- When you are finished, press $\boxed{2\text{nd}}$ $\boxed{\text{QUIT}}$. Press $\boxed{\text{ENTER}}$, then press $\boxed{1}$ or $\boxed{2}$ and $\boxed{\text{ENTER}}$ to replay, or $\boxed{3}$ $\boxed{\text{ENTER}}$ to quit.



$[-9.4, 9.4, 1, -6.2, 6.2, 1]$



$[-9.4, 9.4, 1, -6.2, 6.2, 1]$

Clean-Up

When you are finished running the program, you will probably want to turn the grid off. You should also turn the expressions off. Press $\boxed{2\text{nd}}$ $\boxed{\text{FORMAT}}$ and select GridOff and ExprOff. You might also want to clear an equation that is hidden in the $Y=$ screen in Y_0 .

```
PROGRAM:LINES
ClrHome
0→A
Menu("LINES",
  "EASY",1,"HARD",2,"QUIT",3)
Lbl 2:1→A
Lbl 1
Disp "PRESS ENTER.", "TO SEE GRAPH"
Disp "AFTER YOU ARE","DONE,
  PRESS Y=." :Pause
Func:FnoFF :Plots Off
GridOn:
-9.4→Xmin:9.4→Xmax
1→Hsc1:1→Vsc1
```

```
-6.2→Ymin:6.2→Ymax
ClrHome
Disp "PRESS ENTER","TO CONTINUE."
If A=1:randInt(-9,8)→A
randInt(A+1,9)→C
randInt(-6,6)→B
randInt(-6,6)→D
(D-B)/(C-A)→M
"B+M(X-A)"→Y0
GraphStyle(0,5)
DispGraph
Stop
Lbl 3
" "→Y0
```